

Команди за управление на стандартните потоци за вход и изход на данни

ас. Стоян Мечев

катедра „Информационни технологии“
ВВМУ „Н. Й. Вапцаров“

Въведение

- Поток - последователност от байтове, които могат да бъдат четени или записвани от програми на ОС, като се скриват особеностите на физическото устройство отговарящо за този поток
- Три стандартни потока
 - **stdin** (standard input stream) входни данни за командите.
 - **stdout** (standard output stream) извежда резултатите от командите.
 - **stderr** (standard error stream) извежда съобщенията за грешки дадени в резултат на работата на командите.

| Channel | Name | Abbreviation | Device | Use |
|---------|-----------------------|--------------|----------|--------------------------|
| 0 | standard input | stdin | keyboard | Input for programs |
| 1 | standard output | stdout | screen | Output of programs |
| 2 | standard error output | stderr | screen | Programs' error messages |

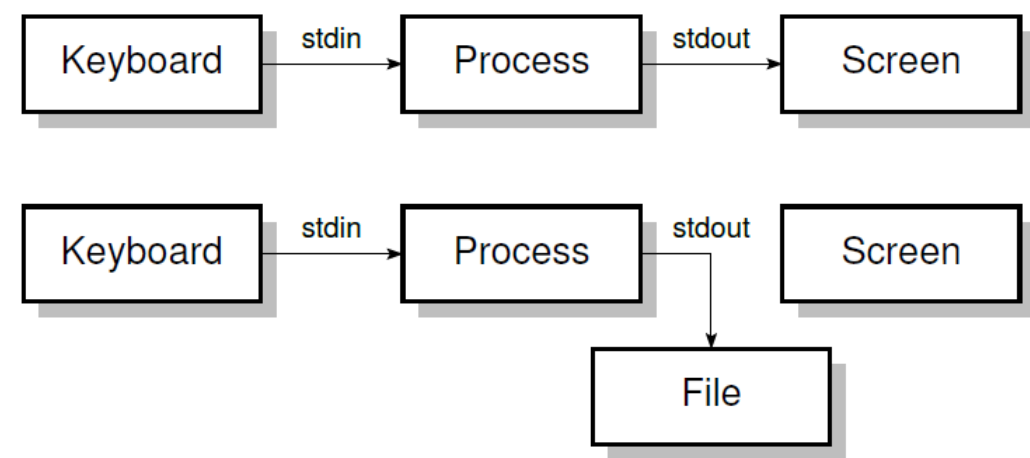
Пренасочвания

- Пренасочване на стандартния изход от една програма към файл ">"

- ако файлът не съществува, той се създава в момента;
- ако файлът съществува, то той се презаписва;
 - За да избегнете презаписването на съществуващ файл, изпълнете командата: `set -o noclobber` в шел.
- **Внимание!** Файл се създава, дори ако командата има грешка и не върне резултат.
- ако се използва ">>" то данните се добавят към файла, без да се изтрива предишното му съдържание.
- Ако се използва "-" означава, че входът идва от stdin, а не от файл
`cat ->some_file ⇔ cat >some_file`

- Пренасочване на файл към стандартния вход на програма;
 - `wc -w <file.txt`
 - `wc -w <file.txt>wordcount` тук пренасочване от файл към програма и резултата към файл.
- Целта, към която се прави пренасочването може да бъде не просто файл, а например принтер или друго устройство;

- Примери:
`ls>text1`
`ping 127.0.0.1>/dev/tty2`



```
ls -y >filelist
ls: неправоплна опция -- 'y'
Try 'ls --help' for more information.
ls -l filelist
-rw-r--r-- 1 user02 user02 0 яну 26 09:16 filelist
```

Пренасочване на един поток към друг

- Използва се операторът '>&'

```
ls -y >filelist 2>&1
```

```
cat filelist
```

```
ls: неправилна опция -- 'y'
```

```
Try 'ls --help' for more information.
```

```
ls -y 2>&1 >filelist
```

```
ls: неправилна опция -- 'y'
```

```
Try 'ls --help' for more information.
```

```
cat filelist
```

Вмъкване на резултата от една команда като част от друга команда (Command substitution).

- Чрез обратен апостроф (backtick) `"`"`
 - `echo "Файлът test.txt съдържа `wc -w <test.txt` думи"`
Файлът test.txt съдържа 2274 думи
 - `echo 'Файлът test.txt съдържа `wc -w <test.txt` думи'`
Файлът test.txt съдържа `wc -w <test.txt` думи
 - обърнете внимание на кавичките
- Чрез субшел
 - `echo Файлът test.txt съдържа $(wc -w <test.txt) думи`
Файлът test.txt съдържа 2274 думи

Команден канал (пайпинг)

Пренасочване на стандартния изход на една програма към стандартния вход на друга

- Свързване на две команди чрез "тръба" - "пайпинг".
 - Не се налага да се занимавате с междинни файлове.
`ls -la|less`
- Резултатът от втората команда може да се насочи отново с `|` към трета и т.н., и отново, крайният резултат може да бъде насочен към файл
 - Няма ограничение за броя на командите, които се свързват в команден канал
- Запазване на междинен резултат във файл: **tee** междинният резултат се записва във файл (с опцията `-a` се добавя)
`ls -la /etc|tee etc_list|less`

Благодаря за вниманието!

Въпроси?